



TITLE:

最小二乗問題に対する内部反復前処理 (科学技術計算における理論と応用の新展開)

AUTHOR(S):

保國, 恵一; 速水, 謙

CITATION:

保國, 恵一 ...[et al]. 最小二乗問題に対する内部反復前処理 (科学技術計算における理論と応用の新展開). 数理解析研究所講究録 2012, 1791: 21-30

ISSUE DATE:

2012-04

URL:

<http://hdl.handle.net/2433/172843>

RIGHT:

最小二乗問題に対する内部反復前処理

保國 恵一[†], 速水 謙^{†‡}

[†] 総合研究大学院大学 (The Graduate University for Advanced Studies)

[‡] 国立情報学研究所 (National Institute of Informatics)

1 はじめに.

優決定最小二乗問題

$$\min_{x \in \mathbf{R}^n} \|b - Ax\|_2, \quad m \geq n \quad (1.1)$$

を解くことを考える. ここで, $A \in \mathbf{R}^{m \times n}$ がランク落ちである場合も許し, $b \in \mathcal{R}(A)$ とは限らないものとする. ただし, $\mathcal{R}(A)$ は A の像空間である. (1.1) は正規方程式

$$A^T A x = A^T b \quad (1.2)$$

と同値である. 劣決定問題 ($m \leq n$) の場合については [15] を参照されたい.

問題 (1.1) に対する直接法には QR 分解や特異値分解がある. 行列 A が大規模疎の場合は, 反復法である CGLS 法 [12] や LSQR 法 [16] が一般に用いられる. しかし, $\kappa_2(A^T A) = \kappa_2(A)^2$ であり, A が悪条件の場合は, それら反復法の収束は遅い. ここで, $\kappa_2(A) = \sigma_{\max}/\sigma_{\min}$ は A の条件数であり, σ_{\max} と σ_{\min} はそれぞれ A の最大および最小の非零の特異値である. 従って, 反復法には収束を加速し記憶容量が少なくすむ前処理が欠かせない [4].

この問題に対する改善法としては, Robust Incomplete Factorization (RIF [3]) 前処理付き GMRES 法 [11] があるが, 計算時間に関して RIF 前処理付き再直交化型 CGLS 法と同等だが, 決定的に優れてはいなかった. さらに RIF は A の非ゼロ要素数に匹敵する記憶容量を必要とし, ランク落ち行列に対して破綻の可能性があった.

最小二乗問題に対する内部反復前処理に関しては, [1] において特異な正方行列を係数行列としてもつ連立一次方程式に対して, 修正した逐次過緩和 (SOR) 法による内部反復前処理を施した一般化共役残差 (GCR) 法が提案されている. しかし, その特異な前処理された系に対して GCR 法の収束が理論的に保証されておらず, 数値実験によっても収束しない例があった.

本論文ではこれらの問題を克服するため, 大規模な最小二乗問題を解くための共役勾配 (CG) 法および一般化最小残差 (GMRES) 法の前処理として定常反復法を内部反復に用いる手法を提案する. 用いる定常反復法は, 正規方程式に対する加速パラメータ付きの Jacobi 法と逐次過緩和 (SOR) 法であり, $A^T A$ を陽に構成する必要がない. この利点は, 従来前処理法 [14, 17, 23, 2, 3, 24, 7, 11, 8] のように前処理行列を陽に構成しないですむことである. その代わりに, 定常反復法を用いて正規方程式を近似的に解くことにより前処理を実現する. 提案手法の目的は, 計算時間と必要記憶容量を小さくし, 適用可能な問題を拡張することである.

本論文の構成を述べる. 第 2 節では内部反復前処理付き GMRES 法と CG 法の枠組みを導入する. 第 3 節では内部反復前処理に用いる具体的な定常反復法を導入し, それらを第 2 節の GMRES 法と CG 法に適用したときの収束性を解析する. 第 4 節では従来法と提案法の比較を数値実験により行う. 第 5 節では本論文をまとめる.

本論文を通じて、ボールド体は列ベクトル、 e_j は単位行列の第 j 列、 (a, b) は a と b の内積 $a^T b$ を表す。内部反復数 k での変数にはカッコ付きの上付き添字を用い (例えば $x^{(k)}$)、外部反復数 k での変数にはカッコなしの下付き添字を用いる (例えば x_k)。

2 内部反復前処理の枠組み.

本節は最小二乗問題に対する CG 法と GMRES 法に内部反復前処理を用いる枠組みを導入する。

[11] により提案された最小二乗問題に対する左前処理付き GMRES は BA-GMRES 法と呼ばれ、 $\min_{x \in \mathbb{R}^n} \|Bb - BAx\|_2$ に対して GMRES 法を適用する。ここで、 $B \in \mathbb{R}^{n \times m}$ は前処理行列である。

B の条件に関する定理を [11] より以下に引用する。ただし、 $\mathcal{R}(A)$ は A の像である。

Theorem 2.1. 任意の $b \in \mathbb{R}^m$ に対して $\min_{x \in \mathbb{R}^n} \|b - Ax\|_2$ と $\min_{x \in \mathbb{R}^n} \|Bb - BAx\|_2$ が同値であるための必要十分条件は $\mathcal{R}(A) = \mathcal{R}(B^T BA)$ である。 \square

Theorem 2.2. $\mathcal{R}(A) = \mathcal{R}(B^T)$ ならば、BA-GMRES 法が任意の $b \in \mathbb{R}^m$ および任意の $x_0 \in \mathbb{R}^n$ に対して破綻することなく $\min_{x \in \mathbb{R}^n} \|b - Ax\|_2$ の最小二乗解を与えるための必要十分条件は $\mathcal{R}(A^T) = \mathcal{R}(B)$ である。 \square

k 反復目の Krylov 部分空間は $\mathcal{K}_k(BA, \tilde{r}_0) = \text{span}\{\tilde{r}_0, BA\tilde{r}_0, \dots, (BA)^{k-1}\tilde{r}_0\}$ となる。ただし、 $\tilde{r}_0 = Br_0$ である。

以下では一般性を失わずに初期近似解を $x_0 = 0$ とし、初期残差は $r_0 = b$ とする。 $x_0 \neq 0$ の場合は、 $r_0 = b - Ax_0$ として $\min_{\tilde{x} \in \mathbb{R}^n} \|Br_0 - BA\tilde{x}\|_2$ の近似解 \tilde{x} を求め、 $x := x_0 + \tilde{x}$ とすればよい。

2.1 内部反復前処理付き BA-GMRES 法.

内部反復前処理付き BA-GMRES 法では、 B を陽には構成せず、ある反復法を何反復か適用することにより、陰的に B を構成し GMRES 法に適用する。その内部反復として例えば定常反復法を用い、その反復数が一定ならば各外部反復で B は変化しない。Algorithm 2.1 は本手法の一般的な枠組みを与える。ただし、 p はリスタート周期、 ε は外部反復の停止条件のしきい値である

Algorithm 2.1. 内部反復前処理付き BA-GMRES(p) 法.

1. Roughly solve $A^T Az = A^T r_0$ to obtain $z \simeq \tilde{r}_0 = Br_0$ by using an iterative method.
2. Compute $\beta = \|\tilde{r}_0\|_2$, $v_1 = \tilde{r}_0/\beta$
3. For $k = 1, 2, \dots, p$, Do
4. Roughly solve $A^T Az = A^T Av_k$ to obtain $z \simeq w_k = BA v_k$ by using an iterative method.
5. For $l = 1, 2, \dots, k$, Do
6. $h_{l,k} = (w_k, v_l)$, $w_k = w_k - h_{l,k} v_l$
7. EndDo
8. $h_{k+1,k} = \|w_k\|_2$, $v_{k+1} = w_k/h_{k+1,k}$
9. Find $y \in \mathbb{R}^k$ that minimizes $\|\beta e_1 - \tilde{H}_k y\|_2 = \|Br_k\|_2$.
10. $x_k = x_0 + [v_1, v_2, \dots, v_k] y_k$
11. If $\|A^T(b - Ax_k)\|_2 < \varepsilon \|A^T b\|_2$, then stop.
12. EndDo
13. $x_0 = x_p$, $r_0 = b - Ax_0$ and go to 1

ここで, $\bar{H}_k \equiv \{h_{pq}\} \in \mathbf{R}^{(k+1) \times k}$ である. 本手法は $h_{k+1,k} = 0$ ならば破綻する. x_k が最小二乗解ならば, 本手法は停止する.

Algorithm 2.1 の 1 行目は, ある最小二乗問題 $\min_{z \in \mathbf{R}^n} \|r_0 - Az\|_2$ もしくはその正規方程式 $A^T Az = A^T r_0$ をある反復法を用いて近似的に解くものと解釈できる. 4 行目も同様である. これを内部反復と呼ぶ. 外部の反復法 (いまの場合では GMRES 法) は外部反復と呼ぶ. 内部反復の最小二乗問題を効率よく近似的に解くために, 正規方程式を解くための簡単な定常反復法を用いる. その際 $A^T A$ を陽に構成しない.

B を固定すると, 各外部反復毎に $\|B r_k\|_2$ を最小化し $x_k \in x_0 + \mathcal{K}_k(BA, \tilde{r}_0)$ を決定する. 右前処理型である FGMRES[18] や GMRESR[21] とは異なり, 本手法は各外部反復ごとに内部反復数を変えられない. 外部反復毎に B を変えると, 第 k 外部反復に対する B を B_k として, 最小化されるのは $\|B_k r_k\|_2$ となる. すると, 近似解 x_k は

$$x_0 + \text{span} \{B_0 r_0, (B_1 A) B_0 r_0, (B_2 A) (B_1 A) B_0 r_0, \dots, (B_k A) (B_{k-1} A) \dots B_0 r_0\}$$

中で探索されるが, これはもはや Krylov 部分空間ではない. 実際, B を外部反復毎に変えた数値実験では本手法は収束しなかったため, 以下では B (内部反復数) を固定する.

2.2 内部反復前処理付き CGLS 法.

同様に内部反復前処理を CGLS 法に対しても適用できる. 以下がそのアルゴリズムである.

Algorithm 2.2. 内部反復前処理付き CGLS 法.

1. Roughly solve $A^T Az = A^T r_0$ to obtain $\tilde{z}_0 = CA^T r_0$ by using an iterative method.
2. $p_0 = \tilde{z}_0$, $s_0 = A^T r_0$, $\gamma_0 = (s_0, \tilde{z}_0)$
3. For $k = 0, 1, \dots$, Do
4. $q_k = Ap_k$
5. $\alpha_k = \gamma_k / (q_k, q_k)$
6. $x_{k+1} = x_k + \alpha_k p_k$
7. If $\|A^T (b - Ax_{k+1})\|_2 < \varepsilon \|A^T b\|_2$, then stop.
8. $r_{k+1} = r_k - \alpha_k q_k$
9. $s_{k+1} = A^T r_{k+1}$
10. Roughly solve $A^T Az = A^T r_{k+1}$ to obtain $\tilde{z}_{k+1} = CA^T r_{k+1}$ by using an iterative method.
11. $\gamma_{k+1} = (s_{k+1}, \tilde{z}_{k+1})$
12. $\beta_k = \gamma_{k+1} / \gamma_k$
13. $p_{k+1} = \tilde{z}_{k+1} + \beta_k p_k$
14. EndDo

1 行目と 10 行目において定常反復法を用いて正規方程式を近似的に解くことで前処理を実現する. ただし, ここで C は正定値対称でなければならない.

3 内部反復前処理.

本節では, 2 節の内部反復前処理に用いる定常反復法を 2 つ導入する. $(k+1)$ 反復目と k 反復目の近似解との差分の第 j 要素を $\delta_j^{(k)}$ とする. つまり $x_j^{(k+1)} = x_j^{(k)} + \delta_j^{(k)}$ とする. 行列 A にはゼロ列がないことを仮定して, $\delta_j^{(k)}$ を

$$(A^T b - A^T A(x^{(k)} + \delta_j^{(k)} e_j), e_j) = 0 \iff \delta_j^{(k)} = \frac{(r^{(k)}, a_j)}{\|a_j\|_2^2} \quad (3.1)$$

により定める.

以下では, 内部反復法の初期近似解は $\mathbf{x}^{(0)} = \mathbf{0}$, \mathbf{a}_j は A の第 j 列とする. ベクトルの要素を非ボールド体で表し, 要素番号に下付き添字を用いる.

3.1 Cimmino-NR 内部反復前処理

Cimmino-NR 法 [6, 19] は (1.2) に対して加速パラメータ付き Jacobi 法を適用したものと同値である. 加速パラメータ ω と (3.1) により k 反復目の近似解ベクトルの要素を一度に更新することで, 以下のアルゴリズムを得る.

Algorithm 3.1. *Cimmino-NR 法.*

1. Let $\mathbf{r}^{(0)} = \mathbf{b}$.
2. For $k = 0, 1, \dots, q$, Do
3. $\boldsymbol{\delta}^{(k)} = D_n A^\top \mathbf{r}^{(k)}$
4. $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \omega \boldsymbol{\delta}^{(k)}$
5. $\mathbf{r}^{(k+1)} = \mathbf{r}^{(k)} - \omega A \boldsymbol{\delta}^{(k)}$
6. EndDo

ただし, $D_n = \text{diag}(1/\|\mathbf{a}_j\|_2)$ である.

本手法を内部反復前処理の観点から解析する. Algorithm 3.1 から, k 反復目の近似解は

$$\mathbf{x}^{(k+1)} = \omega \sum_{l=0}^k (I_n - \omega D_n A^\top A)^l D_n A^\top$$

である. 行列 $B^{(k)}$ を

$$B^{(k)} \equiv \omega \sum_{l=0}^k (I_n - \omega D_n A^\top A)^l D_n A^\top \quad (3.2)$$

と定義すれば $\mathbf{x}^{(k+1)} = B^{(k)} \mathbf{b}$ である. すると $B^{(k)}$ は k 反復の Cimmino-NR 内部反復前処理行列であり, 内部反復付き BA-GMRES 法に適用できる. ここで

$$C^{(k)} \equiv \omega \sum_{l=0}^k (I_n - \omega D_n A^\top A)^l D_n$$

とおけば, (3.2) は $B^{(k)} = C^{(k)} A^\top$ と表せる. D_n は正則であることに注意されたい. 以下では $C^{(k)}$ の (k) を省略する. $\tilde{A} = A D_n^{\frac{1}{2}}$ と定義し, $\tilde{A} = \tilde{U} \tilde{\Sigma} \tilde{V}^\top$ を \tilde{A} の特異値分解とする. すると

$$C = D_n^{\frac{1}{2}} \tilde{V} \text{diag}(\rho_{1,k}, \rho_{2,k}, \dots, \rho_{r,k}, (k+1)\omega, \dots, (k+1)\omega) (D_n^{\frac{1}{2}} \tilde{V})^\top$$

が成り立つ. ここで $\rho_{l,k} \equiv [1 - (1 - \omega \tilde{\sigma}_l^2)^{k+1}] / \tilde{\sigma}_l^2$, $r = \text{rank } A$, $\tilde{\sigma}_1 \geq \tilde{\sigma}_2 \geq \dots \geq \tilde{\sigma}_r > 0$ は \tilde{A} の特異値である. C は対称かつ $\text{diag}(\rho_{1,k}, \rho_{2,k}, \dots, \rho_{r,k}, (k+1)\omega, \dots, (k+1)\omega)$ と合同である. したがって, 以下を得る.

Theorem 3.1.

$$\begin{aligned} \omega < 0 &\implies C \text{ は負定値である,} \\ \omega = 0 &\implies C = 0 \text{ である,} \\ 0 < \omega < \frac{2}{\tilde{\sigma}_1^2} &\implies C \text{ は正定値である,} \\ \frac{2}{\tilde{\sigma}_1^2} \leq \omega &\implies \begin{cases} k \text{ が偶数であれば } C \text{ は正定値である,} \\ k \text{ が奇数であれば } C \text{ は定値ではない.} \end{cases} \quad \square \end{aligned}$$

Theorem 3.2. C が特異であるための必要十分条件は $\omega = 0$, または k が奇数かつある $l, 1 \leq l \leq r$ に対して $\omega = \frac{2}{\sigma_l^2}$ である. \square

Theorem 3.3. C が正則ならば以下が成り立つ. 任意の $\mathbf{b} \in \mathbf{R}^m$ に対して $\min_{\mathbf{x} \in \mathbf{R}^n} \|\mathbf{b} - A\mathbf{x}\|_2$ と $\min_{\mathbf{x} \in \mathbf{R}^n} \|B^{(k)}\mathbf{b} - B^{(k)}A\mathbf{x}\|_2$ が同値である. \square

Theorem 3.4. $\text{rank } A = n$ かつ C が正則ならば以下が成り立つ. BA-GMRES 法が任意の $\mathbf{b} \in \mathbf{R}^m$ に対して 破綻することなく $\min_{\mathbf{x} \in \mathbf{R}^n} \|\mathbf{b} - A\mathbf{x}\|_2$ の最小二乗解を与える. \square

C は対称なので, さらに定値であれば Algorithm 2.2 のように (1.2) を左前処理して CGLS 法を使うことができる.

3.2 NR-SOR 内部反復前処理

NR-SOR 法 [19] は (1.2) に対して SOR 法を適用したものと同値である. ω と (3.1) により k 反復目の近似解を各要素ごとに更新することで, 以下のアルゴリズムを得る.

Algorithm 3.2. NR-SOR 法.

1. Let $\mathbf{r} = \mathbf{b}$.
2. For $k = 0, 1, \dots, q$, Do
3. For $j = 1, 2, \dots, n$, Do
4. $\delta_j^{(k)} = (\mathbf{r}, \mathbf{a}_j) / \|\mathbf{a}_j\|_2^2$
5. $x_j^{(k+1)} = x_j^{(k)} + \omega \delta_j^{(k)}$
6. $\mathbf{r} = \mathbf{r} - \omega \delta_j^{(k)} \mathbf{a}_j$
7. EndDo
8. EndDo

本手法を内部反復前処理の観点から解析する. まず $A^T A = L + D + L^T$ と分離し, $\mathcal{L} \equiv D + \omega L$, $G \equiv I - \omega \mathcal{L}^{-1} A^T A$ とする. ただし, L は狭義下三角行列, D は対角行列である. Algorithm 3.2 から, k 反復目の近似解は

$$\mathbf{x}^{(k+1)} = \omega \sum_{l=0}^k G^l \mathcal{L}^{-1} A^T \mathbf{b}$$

である. k 反復の NR-SOR 法による内部反復前処理行列を

$$B^{(k)} \equiv \omega \sum_{l=0}^k G^l \mathcal{L}^{-1} A^T \quad (3.3)$$

とすると, $\mathbf{x}^{(k+1)} = B^{(k)} \mathbf{b}$ である. $B^{(k)}$ は内部反復付き BA-GMRES 法に適用できる. ここで

$$C^{(k)} \equiv \omega \sum_{l=0}^k G^l \mathcal{L}^{-1},$$

とする. (3.3) から $B^{(k)} = C^{(k)} A^T$ である. 従って, 以下を得る.

Theorem 3.5. $\theta \equiv 2\pi l / (k+1)$, $l \in \mathbf{Z}$ とする. $\text{rank } A = n$ ならば, 以下は同値である.

- $C^{(k)}$ は正則である;

- $(I - G^{k+1})$ は正則である;
- G^{k+1} は固有値 1 をもたない;
- G は固有値に $\exp(\theta i)$ をもたない;
- $[(1 - \exp(\theta i))\mathcal{L} - \omega A^T A]$ は正則である.

ただし, \exp は指数関数, i は虚数単位である. □

Theorem 3.6. $C^{(k)}$ が正則ならば, 以下が成り立つ. 任意の $\mathbf{b} \in \mathbf{R}^m$ に対して $\min_{\mathbf{x} \in \mathbf{R}^n} \|\mathbf{b} - A\mathbf{x}\|_2$ と $\min_{\mathbf{x} \in \mathbf{R}^n} \|B^{(k)}\mathbf{b} - B^{(k)}A\mathbf{x}\|_2$ は同値である. □

Theorem 3.7. $\text{rank } A = n$ かつ $C^{(k)}$ が正則ならば以下が成り立つ. 任意の $\mathbf{b} \in \mathbf{R}^m$ に対して NR-SOR 内部反復前処理付き BA-GMRES 法は $\min_{\mathbf{x} \in \mathbf{R}^n} \|\mathbf{b} - A\mathbf{x}\|_2$ の最小二乗解を与える. □

CGLS 法の前処理には正定値対称な前処理が必要である. Algorithm 3.2 の 4–8 行目の後に $j = n, n-1, \dots, 1$ の順に同様の手続きを行えば NR-SOR 法の対称版である NR-SSOR 法が得られる. [5] は NR-SSOR 法の 1 反復を CGLS 法の前処理に用いる手法を提案した. 提案法ではこれを複数反復できるよう拡張している.

4 数値実験.

3 節の Cimmino-NR 法と NR-SOR 法, NR-SSOR 法を用いて 2 節のように内部反復前処理した BA-GMRES 法および CGLS 法を, 従来の対角スケーリング (diag) および RIF 法により前処理した方法と数値実験により比較する¹. 加えて, NR-SOR 内部反復により前処理した系 $B^{(k)}A\mathbf{x} = B^{(k)}\mathbf{b}$ に対して Bi-CGSTAB 法 [21] を適用した手法も比較する.

第 k (外部) 反復の停止条件は

$$\|A^T(\mathbf{b} - A\mathbf{x}_k)\|_2 < 10^{-6} \|A^T\mathbf{b}\|_2 \quad (4.1)$$

とした. 以下では (4.1) を判定するために \mathbf{x}_k から $A^T(\mathbf{b} - A\mathbf{x}_k)$ を計算する時間は無視する. 内部および外部反復法の初期近似解は $\mathbf{0}$ とした. GMRES 法にはリスタートなし ($p = \infty$) のものを用いた.

行列 A のゼロ行とゼロ列は事前に削除した. ベクトル \mathbf{b} の要素は Fortran 関数 `random_number` によって生成した. ゆえに, $\mathbf{b} \notin \mathcal{R}(A)$ とみなしてよい.

計算機は Dell のワークステーション (Intel Xeon X5492 CPU 3.4 GHz, 16 GB RAM, Scientific Linux 6.1) を用いた. 反復法のプログラムは Fortran 95 言語で書き, コンパイラーは Intel Fortran version 12.1.0 を用いて倍精度演算を行なった. 直接法は MATLAB 2011b の関数 `mldivide` を用いた.

4.1 計算時間の比較.

テスト問題に対する各手法の計算時間の比較を行う. 用いたテスト行列の情報を表 4.1 に示す. “nnz” は非ゼロ要素数, “dens.” は非ゼロ要素の密度, D time は直接法による計算時間, $\|A^T\mathbf{r}\|_2 / \|A^T\mathbf{b}\|_2$ は直接法により得られた解での相対残差を表す. $\text{rank } A$ と $\kappa_2(A)$ は MATLAB 関数 `spnrank` [10] と `svd` により計算した. “-” は計算機の記憶容量不足より計算できなかったことを示す. HIRLAM [13] は [3] でも用いられた² その他の行列は [9] から選んだ. Maragal.8 は $m > n$ となるよう転置した. Maragal.6–8 はランク落ち問題である. Maragal.6 と Maragal.7 を除き, 直接法は (4.1) の要求精度を満たした.

表 4.2 は各問題に対する各手法の計算結果を示す. 各升の一段目は括弧外に (外部) 反復数, 括弧内に計

¹用いた RIF のソースコードは Michele Benzi 教授と Miroslav Tuma 教授により開発され, <http://www2.cs.cas.cz/~tuma/sparslab.html> に公開されている.

²HIRLAM のデータのご提供に関し, Miroslav Tuma 教授と Michele Benzi 教授に感謝する.

算時間が最小であった RIF の閾値、もしくは内部反復数と加速パラメータを示す。各升の二段目は括弧外に総計算時間、括弧内に RIF による前処理行列を計算する時間を示す。最後の列の手法は、 k 反復の NR-SOR 内部反復により前処理した系 $B^{(k)}Ax = B^{(k)}b$ に対して BiCGStab 法 [20] を適用したものである。RIF の最適な閾値は $k \times 10^{-l}$ ($k = 1, 2, \dots, 9, l = 1, 2, \dots, 10$) から選び、 ω の最適値は 0.1, 0.2, \dots , 1.9 から選んだ。直接法の計算時間を各行列名の下に示す。最初の列にある \blacktriangledown は直接法により得られた解が要求精度を満たさなかったことを表す。各問題に対して $*$ は最も計算時間が短かった手法を表す。Maragal.6-8 に対しては RIF 前処理で様々な閾値を試みたが、前処理行列の計算中に破綻したことを Δ で表す。Maragal.6 および 8 に対しては NR-SOR 内部反復前処理付き Bi-CGSTAB 法で様々なパラメータを試みたが、収束しなかったことを \blacktriangle で表す。

HIRLAM に対しては NR-SSOR 内部反復前処理付き CGLS 法が、Maragal.7 に対しては NR-SOR 内部反復前処理付き BA-GMRES 法および Bi-CGSTAB 法が、その他の問題に対しては NR-SOR 内部反復前処理付き BA-GMRES 法が最も短い計算時間で収束した。NR-SOR 内部反復前処理付き BA-GMRES 法は、Maragal.8 に対して特に短い計算時間で収束した。また、全ての問題に対して NR-SOR 内部反復前処理付き BA-GMRES 法は直接法より計算時間が短かった。Maragal.8 に対して対角スケーリング前処理付き CGLS 法は 10^5 反復しても収束しなかった。ところで、これらの前処理を適用した再直交化型 CGLS 法 [11] はここでの結果よりも計算時間が長かった。

図 4.1 は LargeRegFile に対する各手法の相対残差 $\|A^T r_k\|_2 / \|A^T b\|_2$ vs. 計算時間を示す。NR-SOR 内部反復前処理付き BA-GMRES 法が最も速く収束したことが分かる。CGLS 法の相対残差は振動し、収束が遅かった。

図 4.2 は LargeRegFile に対して NR-SOR 内部反復前処理付き BA-GMRES 法が条件 (4.1) を満たすまでに要した計算時間 vs. 加速パラメータ ω を示す。 k は内部反復数である。最小の計算時間を与える最適な ω の値は 1.4、内部反復数 $k = 5$ であった。

表 4.1: テスト行列の情報.

行列名	m	n	nnz	dens. [%]	rank	$\kappa_2(A)$	D time	$\ A^T r\ _2 / \ A^T b\ _2$
Maragal.6	21,251	10,144	537,694	0.25	8,331	2.91×10^6	381.15	1.37×10^{-1}
Maragal.7	46,845	26,525	1,200,537	0.10	20,843	8.98×10^6	483.39	7.95×10^{-2}
Maragal.8	60,845	33,093	1,308,415	0.06	—	—	74.84	1.12×10^{-9}
HIRLAM	1,385,270	452,200	2,713,200	0.0004	—	—	128.97	1.23×10^{-15}
LargeRegFile	2,111,154	801,374	4,944,201	0.00029	—	—	11.16	8.03×10^{-15}

表 4.2: テスト問題に対する各手法の計算結果.

反復法 前処理	CGLS 法			BA-GMRES 法			BiCGStab 法	
	diag	RIF	NR-SSOR	diag	RIF	Cimmino-NR	NR-SOR	NR-SOR
Maragal.6 \blacktriangledown 381.15	11,645 27.83	Δ	2,249 (1, 0.6) 15.29	1,817 42.20	Δ	720 (7, 0.4) 17.43	318 (6, 1.5) * 4.93	\blacktriangle
Maragal.7 \blacktriangledown 483.39	2,822 18.84	Δ	146 (2, 0.9) 4.39	652 17.67	Δ	395 (3, 0.2) 12.71	99 (4, 1.3) * 2.45	71 (3, 1.2) * 2.45
Maragal.8 74.84	$> 10^5$ 847.67	Δ	38,497 (1, 0.7) 757.91	5,376 1,181	Δ	2781, (5, 0.4) 416.42	969 (6, 1.2) * 71.88	\blacktriangle
HIRLAM 128.97	121 6.92	49 (0.2) 4.44 (1.34)	19 (1, 1.8) * 2.31	121 23.28	35 (0.1) 5.55 (2.21)	58 (2, 0.9) 9.30	15 (9, 1.8) 4.01	7 (8, 1.6) 3.26
LargeRegFile 11.20	60 4.67	54 (0.8) 7.99 (3.66)	24 (1, 1.0) 5.21	60 13.23	46 (0.4) 12.65 (3.76)	33 (2, 0.7) 7.27	7 (5, 1.4) * 3.09	9 (3, 1.3) 4.12

1 段目: (外部) 反復数 (内部反復数, RIF の閾値もしくは加速パラメータ)
2 段目: 総計算時間 [秒] (前処理時間 [秒])

4.2 NR-SOR 内部反復の自動パラメータチューニング.

NR-SOR 内部反復前処理では内部反復数 n_{in} と加速パラメータ ω という二つのパラメータが必要である. 図 4.2 で見たように, それらパラメータによって, NR-SOR 内部反復前処理付き BA-GMRES 法の計算時間は変化する. そこで, これらパラメータを問題毎に自動的に決定したい. 定常反復法の最適な加速パラメータの理論値は, 特定の行列に対しては得られている (例えば [25, 22]) もの, 一般の行列に対しては求めることが困難である. そこで, 一般の行列に対して, 外部反復法の開始前に以下の手続きによるパラメータの数値的な最適化法を提案する.

1. Set $\omega := 1$.
2. Starting from $n_{\text{in}} := 0$, find the minimum n_{in} that satisfies

$$\|x^{(n_{\text{in}})} - x^{(n_{\text{in}}+1)}\|_{\infty} \leq \eta \|x^{(n_{\text{in}}+1)}\|_{\infty}.$$

3. Find ω_{opt} that minimizes $\|r^{(n_{\text{in}})}\|_2$.

ここで, η はパラメータ, ω_{opt} は与えられた問題に対する NR-SOR 法 n_{in} 反復での残差を最小化する加速パラメータである. これは NR-SOR 法単体を事前に実行して近似的に最適な n_{in} と ω を決定する方法である. ω_l を ω_{opt} の候補とする. 以下の実験では $l = 19, 18, \dots, 1$ として $\omega_l = l \times 10^{-1}$ から ω_{opt} を決定した.

チューニングされた n_{in} と ω_{opt} は最適値ではないが, その近似となる. $x^{(k)}$ と $r^{(k)}$ は NR-SOR 法の計算過程に現れるものであり, チューニングに要する計算は少なくすむ.

表 4.3 は上記の手続きによりチューニングされたパラメータでの実験結果を示す. 用いたテスト問題は 4.1 の Maragal_6-8 である. 各升の 1 段目は括弧外に外部反復数, 括弧内にチューニングされた内部反復数と加速パラメータを示す. 各升の 2 段目は括弧外に総計算時間, 括弧内にパラメータのチューニング時間を示す. η には 10^{-2} , $10^{-1.5}$, $10^{-1.0}$, $10^{-0.5}$ を用いた. 各問題に対して * は最も計算時間が短かった場合を表す.

チューニングされた NR-SOR 内部反復前処理付き BA-GMRES 法の計算時間は, 表 4.2 での最適なパラメータによる計算時間に近い. また, η が $10^{-1.5}$ から $10^{-0.5}$ の場合の計算時間は, 対角スケール付

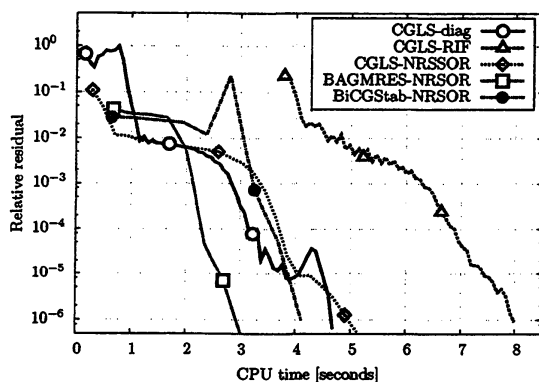


図 4.1: 相対残差 $\|A^T r_k\|_2 / \|A^T b\|_2$ vs. 計算時間 (LargeRegFile).

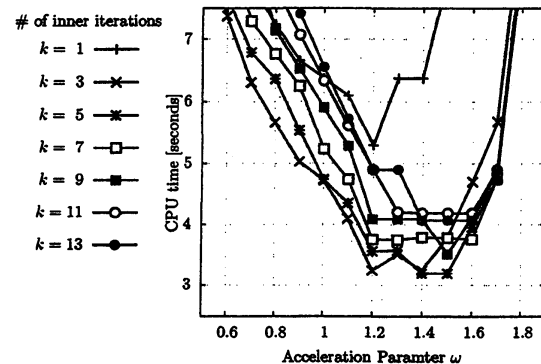


図 4.2: NR-SOR 内部反復前処理付き BA-GMRES 法が (4.1) を満たすまでに要した計算時間 vs. 加速パラメータ ω (LargeRegFile).

表 4.3: パラメータを自動チューニングされた NR-SOR 内部反復前処理付き BA-GMRES 法の計算時間.

η	10^{-2}	$10^{-1.5}$	10^{-1}	$10^{-0.5}$	最適値
Maragal_6	151 (105, 1.9) 27.78 (0.55)	222 (50, 1.9) 19.78 (0.26)	259 (9, 1.6) 5.14 (0.09)	355 (5, 1.4) * 5.00 (0.07)	318 (6, 1.5) 4.93
Maragal_7	102 (46, 1.9) 21.84 (0.62)	144 (13, 1.7) 9.68 (0.29)	165 (7, 1.5) 6.71 (0.22)	99 (4, 1.3) * 2.61 (0.16)	99 (4, 1.3) 2.45
Maragal_8	438 (140, 1.6) 334.80 (4.46)	721 (20, 1.6) 100.27 (0.64)	1,223 (4, 1.3) * 89.82 (0.19)	1,781 (2, 1.1) 150.77 (0.12)	969 (6, 1.2) 71.88

1 段目: 外部反復数 (内部反復数, 加速パラメータ)
2 段目: 総計算時間 [秒] (パラメータチューニング時間 [秒])

き BA-GMRES 法および CGLS 法の計算時間よりも短かった (表 4.2 と比較せよ). さらに, チューニングに要する時間は総計算時間に比べて無視できるほど小さかった. η の値が 10^{-1} および $10^{-0.5}$ である場合に (準) 最適なパラメータが得られた.

5 まとめ.

最小二乗問題に対する Krylov 部分空間法の新しい前処理として内部反復前処理を提案した. 内部反復としては正規方程式に対する加速パラメータ付き Jacobi 法と SOR 法, SSOR 法を利用した. 外部反復として CGLS 法と GMRES 法を用いた.

提案法は, 計算時間と必要記憶容量に関して効率的であり, 特に大規模問題とランク落ち問題に対して効果的である. 定常反復法は一般に収束が遅いが, Krylov 部分空間法の前処理として利用することで速い収束を実現できた. また, これら前処理の理論的保証を与えた.

優決定な大規模問題, またランク落ち問題に対する数値実験から, NR-SOR 内部反復前処理付き BA-GMRES 法が他の手法に計算時間が短いことを示した. また, NR-SOR 内部反復の自動パラメータ調整手法を提案し, その効果を実験的に示した.

参考文献

- [1] D. AOTO, E. ISHIWATA, AND K. ABE, *A variable preconditioned GCR(m) method using the GSOR method for singular and rectangular linear systems*, J. Comput. Appl. Math., 234 (2010), pp. 703–712.
- [2] Z.-Z. BAI, I. S. DUFF, AND A. J. WATHEN, *A class of incomplete orthogonal factorization methods. I: Methods and theories*, BIT, 41 (2001), pp. 53–70.
- [3] M. BENZI AND M. TÛMA, *A robust preconditioner with low memory requirements for large sparse least squares problems*, SIAM J. Sci. Comp., 25 (2003), pp. 499–512.
- [4] Å. BJÖRCK, *Numerical Methods for Least Squares Problems*, SIAM, Philadelphia, 1996.
- [5] Å. BJÖRCK AND T. ELFVING, *Accelerated projection methods for computing pseudoinverse solutions of systems of linear equations*, BIT, 19 (1979), pp. 145–163.
- [6] G. CIMMINO, *Calcolo approssimato per le soluzioni dei sistemi di equazioni lineari*, La Ricerca Scientifica, 2 (1938), pp. 326–333.
- [7] X. CUI AND K. HAYAMI, *Generalized approximate inverse preconditioners for least squares problems*, Japan J. Indust. Appl. Math., 26 (2009), pp. 1–14.

- [8] X. CUI, K. HAYAMI, AND J.-F. YIN, *Greville's method for preconditioning least squares problems*, Adv. Comput. Math., 35 (2011), pp. 243–269.
- [9] T. DAVIS, *The University of Florida Sparse Matrix Collection*, available online at <http://www.cise.ufl.edu/research/sparse/matrices/>, The University of Florida.
- [10] L. FOSTER, *San Jose State University Singular Matrix Database*, available online at <http://www.math.sjsu.edu/singular/matrices/>, San Jose State University.
- [11] K. HAYAMI, J.-F. YIN, AND T. ITO, *GMRES methods for least squares problems*, SIAM J. Matrix Anal. Appl., 31 (2010), pp. 2400–2430.
- [12] M. R. HESTENES AND E. STIEFEL, *Methods of conjugate gradients for solving linear systems*, J. Research Nat. Bur. Standards, 49 (1952), pp. 409–436.
- [13] A. HOLSTAD AND I. LIE, *On the computation of mass conservative wind and vertical velocity fields*, Technical report 141, The Norwegian Meteorological Institute, (2002).
- [14] A. JENNINGS AND M. A. AJIZ, *Incomplete methods for solving $A^T Ax = b$* , SIAM J. Sci. Stat. Comput., 5 (1984), pp. 978–987.
- [15] K. MORIKUNI AND K. HAYAMI, *Inner-iteration Krylov subspace methods for least squares problems*, NII Technical Report, NII-2011-001E (2011), pp. 1–27.
- [16] C. C. PAIGE AND M. A. SAUNDERS, *LSQR: An algorithm for sparse linear equations and sparse least squares*, ACM Trans. Math. Software, 8 (1982), pp. 43–71.
- [17] Y. SAAD, *Preconditioning techniques for nonsymmetric and indefinite linear systems*, J. Comput. Appl. Math., 24 (1988), pp. 89–105.
- [18] ———, *A flexible inner-outer preconditioned GMRES algorithm*, SIAM J. Sci. Comput., 14 (1993), pp. 461–469.
- [19] ———, *Iterative Methods for Sparse Linear Systems*, 2nd ed., SIAM, Philadelphia, 2003.
- [20] H. A. VAN DER VORST, *Bi-CGSTAB: A fast and smoothly converging variant of Bi-CG for the solution of non-symmetric linear systems*, SIAM J. Sci. Stat. Comput., 13 (1992), pp. 631–644.
- [21] H. A. VAN DER VORST AND C. VUIK, *GMRESR: a family of nested GMRES methods*, Numer. Linear Algebra Appl., 1 (1994), pp. 369–386.
- [22] R. S. VARGA, *Matrix Iterative Analysis*, Springer Verlag, 2nd ed., 2000.
- [23] X. WANG, K. A. GALLIVAN, AND R. BRAMLEY, *CIMGS: An incomplete orthogonal factorization preconditioner*, SIAM J. Sci. Comp., 18 (1997), pp. 516–536.
- [24] J.-F. YIN AND K. HAYAMI, *Preconditioned GMRES methods with incomplete Givens orthogonalization method for large sparse least-squares problems*, J. Comput. Appl. Math., 226 (2009), pp. 177–186.
- [25] D. M. YOUNG, *Iterative Solution of Large Linear Systems*, Academic Press, 1971.